

# **A website design primer**

**written by dreamkatcha**

If you believe what you read on the net you'll be well aware that your life is completely empty and meaningless unless you have created a home for yourself on the web. Well, I wouldn't quite go that far, but it's always nice to learn a new skill and the words 'web designer' will look very impressive on your CV (resume to you Americans).

Contrary to popular belief it's really not all that difficult to create and upload your own web site. If you've used Microsoft Word before - almost every computer user has at some point in their lives - you're already halfway there. The stumbling block that tends to hold people back from creating their own web site is that they assume they will have to learn how to write their own HTML code, and often the mere mention of the phrase sends people running for the hills.

For the uninitiated, HTML code is the behind the scenes, bare-bones scaffolding of the web pages you view in your browser, but fear not wannabe web designers, you don't have to know how to construct 'raw code' to create a web page. Many programs will do this for you without leaving tell-tale signs to indicate that you 'cheated'. One exception is Microsoft's Frontpage - a simple to use, Word-like web page builder that produces horrendously convoluted proprietary HTML code which fails to render correctly in many browsers. Avoid it like the black death - at least this plague eventually ran its course! Frontpage is still at large!

# Tools of the trade

To begin with, you aren't going to be interested in writing your site in raw code, that comes much later on when you've mastered the basics. Your first port of call should be to download a 'What You See Is What You Get' (or WYSIWYG for short) HTML editor. One of the simplest and best free ones is [NVU](#). WYSIWYG roughly translates to 'any chump could use this program'. More specifically, such tools allow you to construct impressive-looking documents without ever needing to see the underlying nuts and bolts that generate their contents.

Once you progress from NVU you may wish to apply your newly-acquired skills to Adobe's [Dreamweaver](#), a much more advanced and versatile web page editor. You could even opt to use Dreamweaver from the outset - if you overlook some of the more complicated features to begin with, what remains isn't all that far removed from the NVU interface, and so most of what follows will also apply to Dreamweaver.

Before we delve into the world of web design I want you to know that I'm not an expert at this by any stretch of the imagination. I wouldn't call myself a professional web designer as I've only dabbled, so don't take what I say as gospel. If you want the guidance of a professional refer to the links at the end of this page. What I *can* tell you though is how to get started, which programs to use and how to make things easier for yourself.

OK, let's get this show on the road. Open up NVU and click on the 'new page' icon (actually when you open the program you tend to be confronted with a new blank page by default so this 'arduous' step may be redundant). There are more complicated templates for you to choose from at this stage,

but to begin with we'll keep things simple and take it one step at a time.

When you type the URL of a web site into your browser you are automatically redirected to the default page - this is usually called index.html or index.htm. For instance, if you visit `www.name-of-web-site.com` you will actually be viewing `www.name-of-web-site.com/index.html`, even if the filename doesn't appear in the address bar. For this reason, it makes sense to use the same filename for your first page.

To do this click on the 'save' icon and type in the title you would like to appear in your visitor's web browser whenever they open the page (this will also be the default title for its entry in the favourites folder if someone decides to bookmark it). Now type in 'index' - the html or htm extension will be appended automatically.

It's sensible to create a new directory to group all your pages together, so click on the 'new folder' button and give it a label. Done that? Select the new folder by clicking on it once and press the 'save' button to store your new page in that directory. Once this process is complete you will be brought back to your new empty page where you can begin to add some content.

Now, this is where some experience of using Word comes in very handy. Many of the functions you will come across in Word, and the whole of the Microsoft Office suite for that matter, are standardised so that once you can use one application, you can use them all - see, Bill can get it right *some* of the time! So dive straight in and start filling in the blank space with some text. Any old gibberish will do just to familiarise yourself with the NVU environment. If you look at the top of the screen you will be pleased to discover that all the familiar Word menu bar functions are available. The

most commonly used tweak buttons, those responsible for changing the font of your text, its colour, size and style are never more than a few clicks away.

## Hyperlinks

If you want to create a hyperlink using some of this text, drag your mouse cursor over it while holding down the left button and poke the 'link' icon. If you want to link to an external web site, type its address into the space provided and click on the 'more properties' button. This is here to help you decide how you would like your new page to be opened. The options are fairly straightforward; if you want the site to pop up in a new window so that your visitors can continue browsing your site, tick the 'link is to be opened' checkbox and place a circle in the 'in a new window' radio button.

If you want the linked site to open in the same frame where the hyperlink is situated, select 'in the current frame'. Alternatively, if you want the site to replace your site completely rather than just a single frame select 'in same window, getting rid of all framesets'. If you are planning to use the same format throughout your site, you might like to add into the header section of your pages, the code:

```
<head>  
<base target="_blank">  
</head>
```

In this instance, the code is telling your browser to open all links on a particular page in a new window so that your visitors never have to leave your site - especially handy for links pages because it saves you having to set the target frame for each link. Three other target options are available, "\_parent", "\_self" and "\_top". 'Parent' indicates that the new page should be opened in the parent frameset of the link,

'self' instructs your browser to open the page in the current frame, replacing the content in that frame, and finally, 'top' opens the link in the outermost frameset of the current document, replacing *all* frames.

Much of the time you will want to link to another page of your own site instead of an external link. To do this, it is a good idea to keep all your HTML files together in one directory and make sure that you recreate this directory structure on your remote server as it appears on your own hard drive. To allow your visitors to jump to another page of your site follow the instructions to make a hyperlink as detailed above, but this time click on the 'choose file' button instead of typing in a web site address. Browse to the file on your computer and select it. Providing you tick the 'URL is relative to page location' checkbox the hyperlink will look something like 'anotherpage.html'. There should be no trace of a drive letter, any Ws or an http:// reference.

Doing things this way allows you to edit your pages offline so that they appear instantly instead of being loaded from the webserver in your web page editor. An additional advantage of this method is that no matter where you store your pages, all your internal links will function correctly offline. If at a later date, you have to change hosts, the process simply involves re-uploading the files without having to change any of your links.

If you are linking to a file in another directory of your site the link should look something like 'directoryname/webpage.html'. So you see, it makes no difference where your pages are kept as long as you point to them using the correct directory path without including the web address. From within the same 'link location' panel, a further option available to you is to insert a link to an email address. Whenever one of your visitors clicks on a link

created in this way, their default email client will open a new message window with the selected address already entered into the 'to' field. A handy hint if you want to have the subject field filled in automatically for your visitors is to add "?subject=Your Subject Here" to the end of your email address.

## **Anchors**

If you've got an extremely long page, you may want your visitors to be able to jump to different sections of it without having to scroll manually. The point you jump to is known as an 'anchor'. To insert an anchor, highlight an object or piece of text you would like your visitors to be able to jump to and click on the 'anchor' icon. Give the 'quick jump' link a name and press 'OK' - an anchor symbol will appear alongside the text/object. To link to this section of the page, click on the 'link' icon as before and select the name of the anchor from the drop-down menu. The resulting link will appear in your browser as

<http://www.yoursite.com/index.htm#halfwaydownthepage>.

## **To code or not to code, that is the question**

This would be an opportune moment to share another tip - your ultimate aim in designing your site using a fool-proof program like NVU should be to learn HTML code a bit at a time as you go along. This way you can get started straight away without having to refer to a textbook. As you create objects and insert pictures and text the easy way, spare a few seconds to highlight these elements and click on the 'source' tab to see how they would be created if you were

'raw coding' in a bare-bones text editor such as Notepad or TextEdit.

Soon enough, using this code will become second nature and you can dispense with your web page editor altogether! Before you do though, it's also worth remembering that you can use your web page editor to open any web page on the world wide web to see how they were created. So get used to viewing the source code of other people's web pages as well as your own - note that the emphasis here is on learning from, rather than stealing the work of others.

## **Inserting images**

The next thing you might want to do to liven up your site is to add some images. To do this I would recommend that you create a new directory inside your main web site directory called 'images' or 'pics' and place them inside it. Now prod the 'image' icon, find the picture you want to insert and select it. The picture should magically appear wherever you left the blinking cursor in your NVU window. This can then be moved around the screen by dragging it with the left mouse button. Alternatively, you can centre it, or move it to the extreme left or right using the buttons to the right of the style buttons in the menu bar. If you so wish you can also turn your images into hyperlinks by using the 'link' option as detailed above.

To insert a background picture you would choose 'page colours and background' from the 'format' menu and select a suitable file from your images folder. Alternatively, you can simply pick a solid colour to fill in all that white nothingness.

## **Frames**

As your site expands it's very useful to be able to create a contents page full of links to its various sections. One way of doing this is to split your pages up using frames. You can think of frames as the scaffolding on which you place your pages. A page with frames really consists of several separate HTML pages simultaneously displayed on a single page separated by borders. The advantage of creating a contents page is that it can be made to remain in a fixed position enabling your visitors to navigate your site much more easily.

NVU doesn't include frame-based templates (probably because they are considered outdated and aren't easily indexed by search engines) so if you want to use them you will have to insert the relevant code into your pages manually. Here's some example HTML code you can use to create a page split vertically into two frames...

The resulting page would be comprised of three separate HTML documents - one containing the code above (most likely an index page), a second page containing navigational elements caged in the left-hand frame and a third page containing your site's core material.

If you intend to experiment with frames keep in mind that they aren't the ideal way to construct web pages. This is largely because they don't render well in all browsers/resolutions, individual pages are impossible to browse when opened in isolation of their parent structure and can often look amateurish. Nonetheless, when you're just starting out in web design you won't be expecting to win any awards for your efforts, so if you want to employ frames as an initiation learning exercise, go right ahead.

Out of curiosity, you may like to look at the selection of frame-based templates offered by Dreamweaver. These allow



you to split up your pages in a variety of different ways - for instance, you could have a navigational frame at the top of your index page rather than the side, or even split it into many columns or rows aside from the obligatory contents frame.

Having chosen the configuration that best suits your needs you can proceed to create new pages to go in-between the dividers or insert your existing ones. If you create a navigation menu in one of your narrower frames you can allow your visitors to jump between the different sections of your site. To make sure the pages linked from your navigation menu appear in the larger pane select 'main' from the 'target frame' menu (or cite the frame name manually in NVU), otherwise the new page might replace your menu and the frame will be far too small to accommodate it.

From time to time you may want to insert a frame smack bang in the middle of one of your HTML pages, and having fiddled with the frame templates in Dreamweaver you've probably discovered that neither of these editors are very accommodating when it comes to carrying out this task. This is because these frames, commonly known as 'iframes', are non-standard web page elements and originally weren't 100% compatible with all browsers.

Although this is no longer an issue nowadays, the option to use iframes is still absent from most WYSIWYG HTML editors. Not to worry though, we're not going to be told what we can and can't do by an insubordinate piece of software are we? To get the job done we're going to have to edit the code manually. To do this, place your cursor where you want the frame to appear on your page, flip into 'code view' and enter the following into the space:

```
<iframe br="" iframe="" name="iframe" src="file.htm">  
width="310" height="250" frameborder="0"  
align="center">  
</iframe>
```

The above code is fairly self-explanatory, but just to be sure you know how to customise it to suit your needs let me clarify some of the tag options for you.

The 'iframe name' reference is used to target your new window. For instance, if you call your iframe 'theblackhole', whenever you wish to open a page inside the iframe you would target it in the following manner:  
target="theblackhole". This ensures that your pages won't open over the top of your current page or leap out at you in a new window.

'Iframe src' refers to the initial page you want to appear inside the iframe when you first open the page containing the iframe code.

The width, height and alignment parameters need little explanation.

The frame border option allows you to specify whether or not you would like to add a solid line to your iframe to make it stand out. To ensure the contents of your iframe blend into the page containing the iframe code it is best to set this option to '0' and use the same page background colour for both pages; that way they will look like part of the same page.

Like SSI commands (see below), iframes are a great way of including the main layout elements of your site in each page without having to replicate them in every instance where you want them to appear. Using SSI commands is an ultimately superior way of performing this function, but if

your server doesn't support the parsing of SHTML pages, iframes are the next best solution.

## **Server-side includes (SSI)**

The main drawback with using frames is that they aren't very versatile, i.e. they won't resize automatically to accommodate the differing screen resolutions of your visitors, and this can make your site look very disorganised when viewed using a browser configuration other than the one it was designed for. This and the fact that many search engines have difficulties spidering frame-based web sites has recently led to a move away from their use by professional designers. This isn't to say they have abandoned navigational elements altogether, simply that they are inserting them in a more flexible manner.

For instance, if you wanted a navigation bar to appear at the left-hand side of all the pages of your web site, you would create this menu as an isolated page much like you would if you were planning to use it in a separate frame, but instead of using frames to position the pages, you would issue an SSI command. Whenever a page containing an SSI command is opened in a web browser, the pages 'called' by the command will be incorporated or merged into the page seamlessly as though they were a single file.

Rather than recreating your navigation menu in every single page of your site, you would simply copy and paste the calling command into your pages so that, if at a later date, you decide to edit your navigation menu, you only need to edit a single file. Any changes you make to this file will automatically be reflected in all the subsequent pages containing the SSI command calling your menu. So, as you

can see, using SSI commands can be a tremendous time-saver providing you put in a bit of initial effort.

Before you rush off to start converting your site, you must be aware that not all hosts will allow you to make use of SSI commands, and the ones that do will require a bit of tweaking before they will play ball. As the SSI function is such a nifty party trick, any hosts supporting it will be understandably proud of the fact and will likely want to display this information in a prominent place on their home page.

If you're one of the lucky ones in that your host is SSI enabled, to get started you will need to edit (or create, if you haven't already got one) a .htaccess file (for more information regarding this invaluable miniature marvel see the paragraph relating to creating custom 404 pages below). First of all, open your .htaccess file in a basic text editor such as Notepad and add the following lines to it...

```
AddType text/html .shtml  
AddHandler server-parsed .shtml  
Options Indexes FollowSymLinks Includes
```

If you now upload this file to the root of your web server, the whole of your site will become SSI enabled. Similarly, if you create separate .htaccess files for the different directories of your site you can specify the areas you would like to provide SSI support for, and those you would like to remain SSI disabled. These three lines instruct your webserver to recognise SHTML files as a valid web page format in addition to the more commonly used HTML or HTM formats. The reason for this change is that SSI commands will only be acted upon if they originate from within a file with an SHTML extension. Don't let this format fool you into thinking that SHTML files are anything special; they are merely HTML files

with a different extension - precisely why you can simply edit their extension to render them operational.

Finished that? OK, now we're ready to get down to the nitty-gritty of SSI. Open up the SHTML page you wish to call your navigation menu from and select the 'insert' command from your HTML editor's options menu. Now if you click on the 'server-side include' option you will be whisked away to a browser window from which you can select the file you wish to include. For those of you who want to know what is going on behind the scenes whenever you do this, switch to the code view tab of your editor and see if you can spot changes. The new code should look something like .

Note that previewing the page in your browser will not make it appear as intended; for the page to appear correctly it must first be uploaded to your web server. Another thing to keep in mind is that you must remove any header or footer code or any other styling information from the page you wish to insert, otherwise this code will be 'printed' as plain text creating an unsightly mess. And speaking of unsightly messes, inserted segments of SHTML pages have a habit of doing their own thing unless you restrain them using tables, so if you want to be able to set the location of your inserted elements make sure you place each inserted page in its own table (see the next section).

While we're busy noting things, also note that you can include dynamic script files into your static pages. For instance, if you wanted to place a bulletin board inside a graphical navigation interface you could insert the forum index page of your board into an SHTML page. Whenever a visitor opens one of your forums, the CGI, PHP or ASP page will change from within the SHTML page rather than opening a new page. An added bonus of using such a system is that you can assign your board with an easy to remember

address because the URL in your visitors address bar will remain the same no matter where they roam on your board.

## PHP inserts

Some hosts support PHP, but not SSI. If yours is one of them you could give PHP inserts a whirl instead. They serve exactly the same function as server side includes and are easier to implement. The example code below would allow you to insert a navigational header page into any document with a .php extension:

```
<?php include 'header.php';?>
```

Observe that the code contains the full path to the file to be included, not a relative one or a web address. Use either of the latter and this trick won't work. You can usually find the full path to your web site's root directory listed within your control panel (if you have one) or by searching your host's FAQ documentation. If not, try quoting the folder structure I used in the example above - web hosts often adhere to standardised protocol where folder organisation is concerned so its worth a shot.

This code will only work when inserted into files with a .php extension, though that doesn't mean you have to convert all your pages to PHP before using it. Rename your existing HTML files and you're all set to make use of PHP inserts. You can learn PHP later - for now let's take one step at a time, eh!

### **What looks like an SSI/PHP insert and functions like an SSI/PHP insert, but isn't an SSI/PHP insert?**

There's no punch line, sorry to disappoint you. This is better than a cheap gag so pin back your ears and take note. If

your server doesn't support SSI commands or PHP, but you want to be able to integrate identical headers and footers into your pages without having to code them manually in each page, this is the perfect solution. What you do is code all the repeating elements such as headers and footers and then save them as standalone HTML files. These can subsequently be converted into javascript and inserted into your pages wherever you want them to appear using a call function similar to the one described in the previous sections above.

For instance, if you wanted to insert a standard header into a particular web page you could create an external javascript file called header.js and then insert the following code into the head section of the page where you want the header to appear:

```
<script src="javascripts/header.js" type="text/javascript">
</script>
```

Note that the above example assumes you want to place all your external javascript files in a folder on your webspace account called 'javascripts'. This is merely for the sake of grouping together similar files because they are easier to organise that way. This code informs your browser of the whereabouts of any javascript files you may wish to call elsewhere in the same page. To actually set the location where you want the header to appear you would move your cursor to the relevant section in source code view and enter the following code:

```
<script language="JavaScript">
<!--//
header();
//</script>
```

Once this javascript referencing code has been inserted into each of your pages they can in future be modified by editing a single file rather than you having to replicate the changes in each individual page. This works in a very similar fashion to the SSI commands we discussed earlier, yet the pages will load much faster because they are processed client-side rather than server-side. I've also noticed that web caching seems to be much less problematic when you are dealing with HTML as opposed to SHTML pages, so this is a very welcome additional bonus.

This all sounds far more complicated than it really is, so stick with it and you'll be glad you took the time to get to grips with this little known, but invaluable technique. You'll be relieved to know that the HTML to javascript conversion process is totally automatic. [Javascript Maker](#) does the hard work, all you have to do is tell it where to find the files you wish to convert. The javascript code you must insert into your pages is also automatically generated so all that remains to be done is to copy and paste it into the relevant section of your web creations.

## Tables

Another element you may want to incorporate into your site is a table. Tables are very useful if you need to divide up the text or images in your HTML pages while keeping everything neatly organised and under control. Take your average links page for example. These are often divided into several equal columns into which are placed a string of hyperlinks ordered from A - Z. The reason you can't see any lines in these tables is because the table borders tend to be hidden, but trust me they are there.



Surprisingly enough, to insert a table in NVU you have to click on the 'table' icon. A dialog box will pop up prompting you to decide how many horizontal and vertical cells you would like your table to be divided into. Once that taxing decision has been taken care of, press 'OK' and your new table will appear. By highlighting its cells and right-clicking to bring up a context menu you can enter the properties menu to set its attributes (if you didn't do this when it was first created).

## **URL amnesia**

You may have noticed that it is becoming fashionable to have a shorter, easy to remember web site address. Take for instance the long-winded and ugly URL:  
`http://www.members.somehost.com/personal/yoursite/index.html`

If you registered a much shorter and snazzier redirector like `kickme.to/yoursite` (see the links below for a list of companies who provide this service) you could map it to your site's real location and never again have to cite the windy version. Whenever someone typed this shorter URL into their browser they would automatically be forwarded to your site's actual abode.

This isn't just for aesthetic purposes, it's a very useful way of ensuring that you never lose touch with your visitors. Imagine you've chosen to host your site with a fly-by-night company. When they inevitably ...erm, fly into the night never to be heard from again, anyone who has booked your real URL won't know where to find your site once it re-emerges. By registering a shorter redirector you will always be able to login to your account and update it with your new address. Never again will your treasured virtual home be

banished to the web site graveyard. Instead of advertising your real address you would urge people to bookmark your redirector.

Likewise, if you are planning to submit your site to any search engines you would use your new redirector URL. Having these redirectors visible in your visitor's address bar is the equivalent of advertising for the companies who provide these short URL services so don't be surprised if they actually submit your site to the major web listing directories and search engines themselves saving you the hassle.

Hell, why not really push the boat out and claim a bona fide domain name to assign to your web project? They're as cheap as chips these days if you select the [right registrar](#).

## **Search engines**

...and speaking of submitting your site, whatever you do, don't attempt to do it manually by visiting the home page of each search engine one by one - your wrists will shrivel up and drop off in no time at all (well, maybe you'll just get RSI). Anyway, as Harry Enfield would say, "you don't wanna do that". Nope, what you should do instead is get yourself an automated search engine submission gizmo ('Submit Wolf' and 'Add Web' come highly recommended if you have the cash to spare).

If you're still struggling to get your site noticed you may want to opt for a web-based submission instead. The most thorough ones aren't free of course, but you *can* submit your site to all the major search engines for gratis (see the links below).

## Finding a host

Once you have finished writing your web pages you are going to need a remote host to store them on. There are literally thousands of these available; some are free and others aren't. Most web design guides will direct you towards the more obvious free hosts such as Geocities or Tripod, but I'm not going to because you can do much better. The problem with these hosts is that they will foist advertising banners onto your pages making them look tacky and amateurish. Alternatively, you will have to suffer an annoying pop-up, or an extra frame obscuring a large portion of your pages.

Normally I would recommend stumping up a nominal fee for a basic paid host to avoid all this nonsense, but if you're just starting out and don't know how popular your site is going to be you may as well begin with a freebie and upgrade at a later date if you outgrow the limited features available.

Assuming your ISP account doesn't include a chunk of web space, head over to [www.freewebspace.net](http://www.freewebspace.net) and click on the 'advanced search' button. This is your launchpad to finding the perfect host to suit your requirements. No matter what you're planning to do with your new home on the web, it is imperative that your host offers real FTP support. If you're considering signing up with a host and you see the phrase 'web-based FTP account' plastered across their home page without reference to an additional *real* FTP account you would be well advised to give them a wide birth.

With a web-based file manager you are restricted to processing about five files at once no matter whether you are uploading, moving or deleting and you will not be able to make use of automatic synchronisation programs (more on that later). On the other hand, with your own FTP account

you can automate the whole process of uploading however many files you like while you put your feet up.

At this stage, you are probably wondering how much space your new baby is going to gobble up. Well surprisingly little actually. Unless you're planning to re-write the Encyclopedia Britannica or post hundreds of high-quality photographs, you aren't going to need huge quantities of webspace. I would suggest aiming for around ten megabytes, but obviously, the more you can grab the better up to a certain point.

Any host claiming to offer unlimited space is likely to be highly oversubscribed and will often grind to a halt under the strain. Since you are unlikely to need any more than about twenty megabytes, it's wise not to be too greedy. The same rules apply even if you're putting together a site with lots of downloads since you don't necessarily have to store your goodies on the same host as your HTML files.

Incidentally, the word 'unlimited' is a misnomer anyhow - no host will allow your site to grow exponentially without setting some boundaries where webspace is concerned. If anyone knows where you can buy an unlimited capacity hard drive please let me know. ;)

So with this wish list in mind, you can tell the Free Web Space search engine exactly what it is you're looking for. First of all, make sure you tick the 'FTP' box, select the 'more than 20mb' option from the 'web space' section and select 'no ads' from beneath the 'ads' heading. Now click on the search button and commence thumb twiddling until the results appear. You should be presented with a minuscule list of maybe three or four hosts. That's not a problem - after all, you're asking for a lot from a free service.

Armed with your shortlist visit the home page of each of these hosts and select the one offering the best deal. One other thing you may want to take into consideration - as a general rule of thumb, a home page that looks like it's been thrown together by your three-year-old niece or nephew reflects very badly on the reliability of the service. If the site is very difficult to navigate and is lacking in technical support you will often find that many other key services are also being neglected, so shop around before making your final decision.

Whichever host you choose you will have to fill in a few forms before you can start uploading. This involves choosing a username and password as well as handing over a few personal details such as your real name, address, phone number and so on. You *could* enter absolutely anything here if your privacy is a top priority, but if you do provide phoney contact information don't be surprised if your host vaporises your account for violating their terms of service agreement.

Providing your details are accepted you will be sent an email (instantaneously in many cases) containing all the necessary information you will need to upload your pages. It is very important that you do not forget these details, so for future reference make sure you save the email and keep it in a safe place.

## **Uploading your site**

To get your web site from your desktop onto the world wide web you will need to get hold of an FTP client. If you aren't familiar with how these work refer back to my FTP tutorial. All the login details you need to gain access to your host's server should have been included in the confirmation email you were sent. What you will need is... an empty washing up

bottle, a yoghurt pot and... oops, I must have slipped into Blue Peter presenter mode for a second there.

Ahem... your username and password and the FTP address. Enter these details into the relevant fields of your FTP client and connect to the server. Once access has been granted you can begin uploading your pages. To do this enter the directory where your offline site is located and highlight all of its contents including the sub-directories, drag it into the 'files to upload' window and hit the go button.

This method is fine if you rarely update your pages, but if you start making changes left, right and centre you will quickly lose track of the pages you have to re-upload. This is where the miraculous Crystal Site Updater comes to the rescue. By pointing this tiny program towards your offline site it will compare it with the online version and track any changes between the two. If it finds that you have updated, deleted, added or renamed any of the contents of your web site since the last time you connected to the server it will automatically delete the old files from your server and re-upload the new versions in their place so they are always in synch.

If you find that the functions offered by CSU are too simplistic for your needs make sure you look into the web site management system implemented by Adobe's Dreamweaver.

Once the process is complete, you can disconnect and check that everything has gone according to plan by typing the URL of your site into your browser's address bar. Internet Explorer or Netscape Navigator should then locate your index file automatically and display it for you in all its glory. Congratulations, you are now officially a webmaster and

your site will be available to anyone with internet access anywhere in the world! It feels good, doesn't it?

## And now for something a bit more impressive...

First up, **CSS rollover effects**. No, these have nothing to do with teaching your dog to do tricks. A rollover effect is a dynamic web page element, composed of either text or images, that changes as you hover your mouse pointer over it. NVU doesn't yet support easily implemented rollover effects, so if you want to add one to your page you will have to do it manually. Here's a simple example...

```
<style type="text/css">
<[!--
a:link, a:visited { text-decoration: none; }
a:hover, a:active { text-decoration: underline; }
</style>
```

Try inserting this code into the header section of one of your pages and open it in your favourite web browser. Observe what happens when you linger over a text link with your cursor and then move it away again. In this case both 'unmolested' (for want of a better word) and previously visited links are undecorated, whilst hovered-over links and those currently being tampered with i.e. in mid-click mode are underlined. Here the four modes (link, visited, hover and active) have been combined into two groups to produce a subtle rollover effect, though could just as easily be referenced individually if you wished to assign a different style to each one.

The code below will produce a slightly different rollover effect...

```
<style type="text/css">
<!--
a:link { color: #FF0000; text-decoration: underline;}
a:hover { color: #0000FF; text-decoration: none;}
</style>
```

In this instance, whenever a link is hovered over, it changes from underlined, red text to blue text minus the underline.

Other style attributes you can modify to produce interesting rollover effects include 'font-family', 'font-style' (e.g. italic, bold), 'font-variant' (e.g. upper or lower case) and 'font-size'.

Creating CSS rollover effects using images is a bit more complicated, but well worth the extra effort when you see the fruits of your labour. Very often people choose to use image rollovers in conjunction with navigation buttons to give the impression that their visitors are physically interacting with them. For example, you could create a 'home' button using two images, one embossed and one depressed, so that when your visitors hover their cursor over it they get the impression they are pushing the button into its socket. You can create these images in any old paint program or use an online button creation service (see the links below). Once your images are web-ready, visit [this lengthy tutorial](#) to discover where you go from here.

If you haven't got the stamina to complete all these steps, you might like to investigate creating image rollovers using dynamic HTML effects or javascript. The latter techniques literally take seconds to master though are frowned upon by more experienced webmasters as they introduce a number of frustrating caveats such as the necessity for pre-loading and multiple images. The choice is yours.



How about inserting a **java applet** or two into your pages? The first thing to note is that java and javascript are two totally independent languages. Javascript is a much less sophisticated language and is used primarily to create simple web page elements like (old fashioned versions of) the rollover effects already discussed, or drop-down menus; [the possibilities are endless](#). Javascript is not dissimilar to HTML in that you can type its raw code into the 'behind the scenes' area of a web page to create all sorts of interesting effects. This isn't just art for art's sake, it can be used for all sorts of functional purposes as well.

Java, on the other hand, is a programming language used to create applications known as applets, which can either be used offline (or at least independently of your web browser) or by embedding them into web pages. Examples include chat applets, virtual hard drive interfaces and file-sharing clients.

Unless you're a masochist, or you're getting paid for it, there is no reason you should even attempt to write your own javascript code or java applets. Take it from me, it's a very laborious and lengthy procedure and is best avoided at all costs. Luckily for us, every bit of javascript and every java applet you could possibly want to implement on your site has already been slavishly coded by web script professionals and is available for free providing you know where to look (again see the links below).

To get started, visit a free javascript/java site and find an applet or script you'd like to adopt. Click on its link and read through the instructions. You will quickly realise that this couldn't get any simpler; you merely have to copy the code from the web page using your right mouse button and paste it into the HTML pane of your web site editor under the relevant tag. Note that some applets or scripts will require

you to upload a number of essential accompanying files to your web server before they will function correctly. Finally, skip to the preview pane and watch what happens. Hopefully, you haven't unleashed a monster upon your unsuspecting visitors!

If you've ever wondered how those tiny favourites icons are created, wonder no more. All you have to do is design a 16 x 16 GIF image, save it as `favicon.ico` and upload it to the root directory of your web server. Whenever someone bookmarks your site this picture will appear alongside the title of your page in their favourites menu.

If you'd like to treat your visitors to a bit of eye candy while they're getting lost in your dead links you can create your own customised 404 page. Bear in mind that this only works if your host allows you to edit the `.htaccess` file in the root of your web site. Most free hosts don't, so it helps if you either own the server your site is stored on, or you are paying to lease a small chunk of it. You'll be delighted to know this requires only a minor, very quick and simple tweak.

Begin by creating a new document in Notepad and type "ErrorDocument 404 <http://www.yourhost.com/404.html>" onto the top line. Now save the file as `.htaccess` and place it alongside your HTML files (note that this file should *not* have an extension - if Notepad gives it the suffix `.txt` you will have to remove it later, otherwise it won't work). This time open your HTML editor and create a new page called `404.html`, type something witty and original onto it (refer to the [404 Research Lab](#) if inspiration isn't forthcoming) and upload both files to your web site.

In future, whenever someone clicks on the link to a file that no longer exists, instead of seeing a bog-standard, default

404 page they will be greeted with your new funky 'you've lost your way' message.

## **Have a break, have a Kit-Kat!**

It feels as though I've been rabbiting on for an eternity and yet have still only managed to provide a rudimentary web design prologue. There remain many more complicated aspects of web design (such as Flash, Photoshop and the like) I've not even begun to touch upon, but by this stage, you should at least have a good idea of how to get started and where to look for additional assistance. The rest you can learn as you go along, one step at a time. After all, there is no substitute for the hands-on approach, so roll up your sleeves and get stuck in. Don't be scared to experiment with all the options NVU (and later Dreamweaver) has to offer - the rest will fall into place naturally. Have fun!

-----

**As promised throughout this tutorial, what follows is an eclectic mix of useful web design links. Any that have since died over the years since writing this have been deactivated.**

[100 Best Free Web Space Services](#) - Fairly self-explanatory this one.

[1001 Fonts](#) - Go on, guess what this site is all about.

Add Engine - Allows you to submit your site to 25+ major search engines for free.

Add Me - Submit your site to the top 30 most popular search engines for free.

Anfy - The best program available for creating 'off the peg' special effect and navigational java applets.

Big Nose Bird - Free CGI scripts, graphics and tutorials (some of the easiest to follow on the web I might add).

[Boogie Jack](#) - Free graphics and HTML tutorials.

[Brain Jar](#) - Web scripting at its very best.

Cjb.net | Kickme.to | V3 | IP Fox - Free, easy to remember redirector URLs.

[DHTML Shock](#) - Free DHTML scripts.

[Dynamic Drive](#) - An excellent sources of ready-made javascript snippets.

[Echo Echo](#) - More tutorials to guide you through every stage of building your first web site. Also includes a selection of online tools, free resources and forums.

[Flaming Text](#) | [Cool Text](#) - A quick and easy way to create impressive-looking logos and buttons.

[Flash Kit](#) -The no. 1 place to visit if you want to get to grips with Macromedia's Flash.

Free Image - A computer graphics community and resource site.

[Free Web Templates](#) - No prizes for guessing what this one's for.

Granite Canyon | [My Domain](#) | Every DNS | [Indian Magic](#) | [Zone Edit](#) | [Easy DNS](#) - Free DNS providers.

[HTML Goodies](#) - Lots of expertly written tutorials for both the beginner and more advanced web designer.

[Netcraft Web Site Finder](#) - Similar to performing a [Who Is](#) lookup but lists many more complex statistics such as server downtime percentages and whether the host servers are running on a Unix or NT platform (essential information when you're setting up a bulletin board).

Net Mechanic - Free tune-ups and web site analysis.

[Pixel 2 Life](#) - Excellent source of web design tutorials covering topics as diverse as Photoshop, HTML, Bryce, Flash, PHP and Poser.

[Planet Source Code](#) - One for the more adventurous web designer.

[Team Photoshop](#) - Lots of exceptionally well written and easy to follow Photoshop tutorials.

[W3 Schools](#) - The defacto standard for web site construction tutorials.

[Web Page Tools](#) - Free ASP, HTML and Javascript code generators and resources.

[Xara](#) - Home of Xara 3D and Xara Web Style, two of the best template-based graphics creation programs around.

Zy - Make logos, buttons and web templates the easy way.